

# Setting Up a Natural RPC Environment

To set up a Natural RPC environment, you must perform the following steps for all client and server Naturals:

- Setting Up a Natural Client
  - Setting Up a Natural Server
  - Setting Up an EntireX Broker Access
  - Setting Up an EntireX Broker Environment
  - Starting a Natural Server
  - Considerations for Natural RPC Servers with Replicates
- 

## Setting Up a Natural Client

To set up a Natural client proceed as described below:

1. Define the name of the server to be used.

Use the Service Directory Maintenance (under Natural RPC 5.1 or under Natural RPC 3.1) function of the SYSRPC utility and define the name of the server to be used for each CALLNAT to be executed remotely. The generated directory subprogram NATCLTGS must be made available to the Natural client application. If you have not generated NATCLTGS in your client library, you have to move NATCLTGS to this library or to one of the Steplibs.

Optionally, you can use the following server selection techniques:

- Address a default server;  
for more information, see Specifying a Default Server Address Dynamically, or profile parameter DFS.
- Try alternative servers;  
for more information, see Modifying RPC Profile Parameters Dynamically, or profile parameter TRYALT.
- Use a Remote Directory Server (RDS),  
for more information, see Using a Remote Directory Server, or profile parameter RDS.

### **For Windows, OpenVMS and UNIX Environments:**

Predict servers are not maintained in the SYSRPC utility. For information on how to connect to a Predict server, see the profile parameter USEDIC or the Dictionary Server Assignments function in the Global Configuration File.

2. Generate a stub subprogram.

Skip this step, if you want to work without stub. In this case, set the Natural profile parameter AUTORPC to ON, see Working with Automatic Natural RPC Execution.

For each CALLNAT to be executed remotely, use the Stub Generation function of the SYSRPC utility, see Creating Stub Subprograms.

The generated stub must be made available to the Natural client environment. If you have not generated the stub subprogram in your client library, you have to move the stub subprogram to this library or to one of the Steplibs.

3. Set the Natural profile parameters relevant to the client-specific handling of remote procedure calls.

These parameters are (all optional, except RPCSIZE on mainframe clients):

RPCSIZE, MAXBUFF, TIMEOUT, AUTORPC, TRYALT, ACIPATT, CSCPATT (for OpenVMS only), COMPR, DFS, RDS.

## Setting Up A Natural Server

A Natural server is a Natural task that can execute Natural subprograms (services). This Natural task is typically an asynchronous or background task (detached process). The EntireX Broker and the client identify it by using a *nodename* and a *servername*.

To set up a Natural server proceed as described below:

1. Set the Natural profile parameters relevant to the general and server-specific handling of remote procedure calls in a parameter module for the server NATURAL.

The mandatory profile parameters are:

SERVER, SRVNAME, SRVNODE, RPCSIZE (RPCSIZE refers to mainframe servers only).

Optional parameters are:

RPCSIZE, MAXBUFF, TIMEOUT, LOGONRQ, SRVUSER, TRANSP, TRACE, ACIVERS and CP.

If the EntireX Broker is used, the name specified with SRVNODE must identify an active EntireX Broker and the name specified with SRVNAME must match a server definition in the EntireX Broker Attribute File, see Setting Up an EntireX Broker Environment

### **For Mainframe Environments:**

If you want to use TCP/IP, you are recommended to set the TRANSP parameter accordingly, as the preferred transport method is using Entire Net-work.

2. Ensure that your Natural server session will enter command mode:
  - Set MENU=OFF in your Natural profile parameters.
  - Do not put a program onto the Natural stack which never terminates.
  - Do not use a STARTUP program which never terminates.
  - Do not disallow NEXT mode in Natural Security for your server library.
3. Ensure that the ADABAS ETID used by the Natural server session is unique within a certain Adabas nucleus.
4. Start a Natural server as described in the section **Starting A Server** below. This server then waits for remote CALLNAT requests from a client.

### **For OS/390 and VSE/ESA in batch mode:**

For information about servers using the NTASK parameter, refer to Considerations for Natural RPC Servers with Replicates.

## Setting Up an EntireX Broker Access

To set up an EntireX Broker interface proceed as follows:

1. Make the EntireX Broker stub accessible to your Natural environment.

**For Mainframe Environments:**

- If you use the Entire Net-work protocol:  
Link the EntireX Broker stub NATETB23 to your Natural or specify RCA=BROKER to load NATETB23 dynamically at run-time.
- If you use the TCP/IP protocol:  
Specify RCA=BROKER RCALIAS=(BROKER,*stubname*)  
where *stubname* refers to one of the TCP/IP-enabled EntireX Broker stubs BKIMBTSO, BKIMBTIA, EXAAPSB or EXAAPSC.

**Attention for Natural RPC Version 5.1:**

In OS/390 batch mode and under TSO, you may either use BKIMBTSO or EXAAPSB.

You must use the SMARTS-based EXAAPSB:

- If you want to communicate with the EntireX Broker using SSL.
- If you want to use the location transparency provided by the EntireX Broker.

Refer to the EntireX documentation for details.

**For UNIX:**

The EntireX Broker library stub can be assigned in the Local Configuration File of the Natural parameter module, entry NATEXTLIB.

**For Windows:**

The EntireX Broker stub must be accessible over the registry.

2. Set the RPC parameter ACIVERS according to your requirements:

**Note:** The ACIVERS value set in the parameter module can only work if the EntireX Broker and EntireX Broker stub support this version as well.

Setting	Function
<b>ACIVERS=2</b>	(Default) Support of the EntireX Broker functions LOGON and LOGOFF. With Natural Version 3.1 (for mainframes) and Version 4.1 (for Windows, OpenVMS and UNIX environments), the server performs a LOGON to the EntireX Broker before executing the REGISTER, and a LOGOFF after the Deregister. This does not imply any security checks, but it is a pure EntireX Broker management function, see EntireX Broker function LOGON.
<b>ACIVERS=3</b>	Support of EntireX Broker non-numeric conversation IDs. When this Natural parameter is set to 3 or higher, the EntireX Broker will also assign non-numeric conversation IDs. If a Natural client issues an OPEN CONVERSATION and the client's ACIVERS is 3 or higher, the EntireX Broker will be able to automatically assign non-numeric conversation IDs. It will not check whether the associated server does accept non-numeric conversation IDs, but only the ACIVERS of the requestor (a Natural client in this case) will be decisive. <b>Therefore, make sure that both the Natural client and the server support the respective ACI version.</b>
<b>ACIVERS=4</b>	Support of code pages and (for servers only) Natural Security. With EntireX Broker ACI Version 4 or higher, the Natural RPC supports code pages. For this, the name of the code page can be specified in the Natural profile parameter CP for clients and servers. The evaluation of the code page is done by the EntireX Broker. The EntireX Broker translates the RPC data sent according to the code page of client and server to the corresponding target code page. The CP parameter can be set by the client and/or by the server. It applies for the current process. This means that the client code page does not need to be identical with the server code page. With Natural Version 3.1 (for mainframes) and Version 4.1 (for Windows, OpenVMS and UNIX environments), the server is enabled to logon to the EntireX Broker using a qualified user ID. If the Natural parameter/subparameter SRVUSER is set to *NSC and the server is running under Natural Security, the Natural RPC will automatically pass the current Natural user ID (*USER) and the password defined in Natural Security to the EntireX Broker, where they are checked for conformity with the EntireX Broker security data.
<b>ACIVERS=6</b>	If you are using the EntireX Broker stub EXAAPSC (CICS only), we strongly recommend that you use the specification of the ACI Version 6. In this case, Natural will use the TERMINATE option for the LOGOFF from the EntireX Broker.

- For additional RPC parameters affecting the EntireX Broker, refer to the Profile Parameters section in the Parameter Reference documentation.

## Using TCP/IP as a Transport Method

If TCP/IP is used as transport method, you must define the server node in the **hosts** and **services** directory of your TCP/IP installation, as the length of the profile parameter *srvnode* is restricted to 8 characters which disables the use of full TCP/IP addresses with port numbers.

This restriction is obsolete with Natural RPC Version 5.1 as the length of the server node can be up to 32 characters.

## Setting Up an EntireX Broker Environment

In the EntireX Broker Attribute File, add the following:

1. For each Natural RPC server, a service definition must be specified as follows:  
CLASS=RPC, SERVICE=CALLNAT, SERVER=*servername*.
2. If you want to use the conversion services, set CONVERSION = *userexit*. In this case, you must set the Natural profile parameter CP accordingly.
3. If the Natural RPC client and the Natural RPC server are of Natural Version 3.1 or higher (mainframe environments) or of Natural Version 4.1 or higher (Windows, OpenVMS and UNIX environments), you can set AUTOLOGON=NO.  
In this case, ACIVERS must be 2 or higher.
4. If both the Natural RPC client and the Natural RPC server are of Natural Version 3.1 or higher (mainframe environments) or of Natural Version 4.1 or higher (Windows, OpenVMS and UNIX environments) and Natural Security is installed, you can enable EntireX Security by setting:  
SECURITY=YES.  
In this case, SRVUSER must be set to \*NSC on the server side.

## Starting a Natural Server

Any kind of Natural session can be used as a Natural RPC server. But typically, a Natural server is a Natural session which is started as an asynchronous or as a background task.

For the purpose of starting a server, you have the following options:

- Create an RPC-specific Natural parameter module, see Setting Up a Natural Server.
- Alternatively, you can also specify the profile parameters dynamically (PARM=*parm*).
- **In mainframe environments only:** The RPC parameter may be specified in a profile created with the SYSPARM utility.  
Natural would then be started with  
PROFILE = *srvprof*  
where *srvprof* is the name of the profile.

How a Natural server is started depends on the environment.

## Starting a Natural Server in a Mainframe Online Environment

To start a Natural server in a mainframe online environment, enter the following command:

```
<Natural transaction code>
  RPC=(SERVER=ON, SRVNAME=servername, SRVNODE=nodename,
        RPCSIZE=n, MAXBUFF=n)
```

### For CICS and Com-plete only:

You can also use the Natural program STARTSRV in library SYSRPC to start a Natural server in asynchronous mode.

## Starting a Batch Server

Batch servers are started correspondingly, for a sample JCL see Using the Server Trace Facility.

You can also run a batch server with replicates by setting the RPC parameter NTASKS to a value greater than 1 (OS/390 and VSE/ESA only). Replicates are attached to a Natural main task as additional server tasks. They enable you to start several identical servers in the same region.

## Starting a Server in an OpenVMS Environment

To start a Natural server under OpenVMS, enter the following commands in the DCL command procedure "*myserver.com*":

```
$ DEFINE NATOUTPUT NLA0:
$ NAT parm=serverparm
```

*serverparm* must be entered in upper-case letters.

Then submit "*myserver.com*" to a batch queue: \$ SUBMIT *myserver.com*

### Note:

If you are using Natural for OpenVMS Version 2.1.7 as a client or as a server in an RPC conversation, and you are using CSCI, the server-name must contain the "#" character, for example: "SRV#1".

## Starting a Server in a UNIX Environment

To start a Natural server under UNIX, enter the following command:

```
natural parm=serverparm >/dev/null <dev/null &
```

## Starting a Server in a Windows Environment

To start a Natural server under Windows, proceed as follows:

1. Create a shortcut for Natural.
2. Enter the shortcut properties.
3. Create a Natural parameter module with the RPC server parameters set.
4. In the Target input field, edit the Natural path and append:

```
PARM = serverparm batch
```

# Considerations for Natural RPC Servers with Replicates

For OS/390 and VSE/ESA only.

## Natural RPC Batch Server with NTASKS >1

The main task and all replicates run in the same OS/390 region or VSE/ESA partition.

1. Use the reentrant version ADALNKR of the Adabas link module ADALNK.  
If you use ADAUSER, you must rename ADALNKR to ADALNK.  
**Note:**  
You may need a separate Copy of the reentrant ADALNK module if you are using 3GL programs which do not pass a work area as 7th Adabas parameter to the Adabas interface.
2. In the NATPARM module:
  - Set the NTRPC subparameter NTASKS =  $n$ , where  $n$  is the number of parallel servers (< 100) to be started, including the main task.  
**Note for VSE/ESA:**  
The number of subtasks is restricted by the operating system. Ask your system administrator.
  - ETID must be specified as a **blank** character to prevent a NAT3048 (ETID not unique in Adabas nucleus) error when the subtask is started.
3. When using dynamic Natural profile parameters:  
Use the CMPRMIN dataset to pass the dynamic Natural profile parameters to Natural. Do **not** use the PARM card or the CMSYNIN dataset.
4. When using a local buffer pool (OS/390 only):  
Each subtask allocates its own local buffer pool unless you specify a shared local buffer pool. See Natural profile parameter LBPNAME in the section NTOS Macro - Generation Parameters for Natural under OS/390 in the Natural Operations for Mainframes documentation.
5. In the Natural front-end link job (OS/390 only):  
Link the front-end reentrant by using the RENT option of the linkage editor.  
If the front-end is not linked with the RENT option, only the main task will start the communication with the EntireX Broker. All subtasks are set to a WAIT status by OS/390, until the main task has been terminated. If you terminate the RPC server lateron, the address space will hang and must be cancelled.  
**Note:**  
If you use ADAUSER you must not link ADAUSER with your front-end as ADAUSER is non-reentrant. Instead, use the Natural profile parameter ADANAME and set ADANAME to ADAUSER. This will cause Natural to load ADAUSER dynamically at runtime.
6. Make sure that any other modules that are additionally linked to the Natural nucleus are reentrant. Any dynamically loaded programs must also be reentrant.  
**Note for OS/390:**  
If you cannot make a module reentrant, link the module as non-reusable (do **not** specify the link option RENT or REUS). This will ensure that each subtask gets its own copy.

## Running a Batch Server with Replicates

For a sample JCL, see Using the Server Trace Facility.